(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 0 973 297 A2

(12) EUROPEAN PATENT APPLICATION

(43) Date of publication:
19.01.2000 Bulletin 2000/03

(51) Int Cl.[7]: **H04L 12/26**, H04L 29/06

(21) Application number: **99305377.6**

(22) Date of filing: **06.07.1999**

(84) Designated Contracting States:
AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU MC NL PT SE
Designated Extension States:
AL LT LV MK RO SI

(30) Priority: **17.07.1998 US 118425**

(71) Applicant: **TEKTRONIX, INC.**
Wilsonville, Oregon 97070-1000 (US)

(72) Inventor: **Gauland, Michael A.**
Hillsboro, Oregon 97123 (US)

(74) Representative: **Burke, Steven David et al**
R.G.C. Jenkins & Co.
26 Caxton Street
London SW1H 0RJ (GB)

(54) Web based remote monitoring/control of a measurement instrument

(57) A measurement instrument is web enabled, enabling it to receive commands or information to effectuate an operation on the measurement instrument from a user using a browser executing on a remote client computer. In response, the operation is effectuated in accordance with the commands or information provided. Results of the operation, where applicable, are returned to the browser of the remote client computer for the user. The operation may be any one of a number of monitoring or control operations, and the commands or information facilitate identification of the monitoring or control operation of interest. The results of the operation may include data obtained from the measurement instrument.

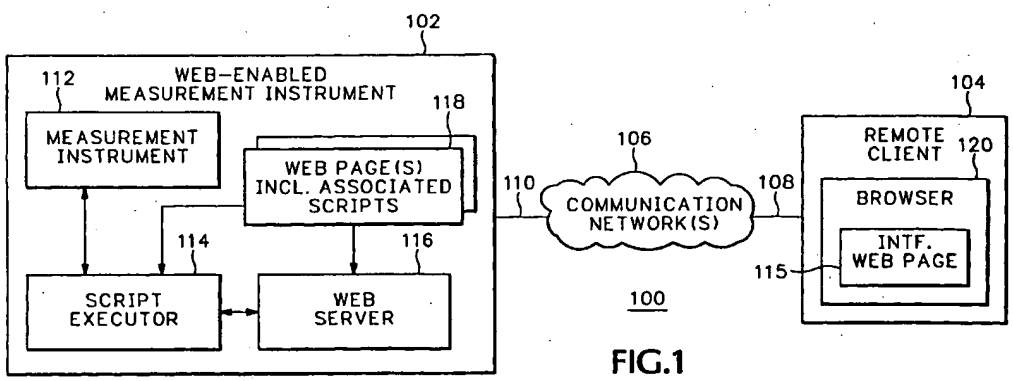


FIG.1

EP 0 973 297 A2

Description

Field of the Invention

[0001] The present invention relates to the field of measurement instruments. In particular, the present invention relates to monitoring or control of a measurement instrument.

Background of the Invention

[0002] Various measurement instruments including oscilloscopes are known in the art. Measurement instruments are used in a wide variety of applications, from measuring engine vibrations to measuring brain waves, just to name a couple of examples. Conventional measurement instruments all suffer from at least the following disadvantages:

(a) monitoring or controlling from a proximity close to the location where measurements are being taken is typically required, and
(b) for remote monitoring or control, e.g. in space explorations, proprietary interfaces or mechanism must be specifically provided.

[0003] Since it is often desirable in routine earthly usage of measurement instruments to monitor or control a measurement instrument from a remote location physically separated from the location where measurements are being taken, e.g. from an office next door, from an adjacent building, or from a different plant site, it is desirable to have a more easily enabled and more user friendly approach to facilitate the desired remote monitoring or control. As will be disclosed in more detail below, the present invention achieves these and other desirable results.

SUMMARY OF THE INVENTION

[0004] A measurement instrument is web enabled, enabling it to receive commands or information to effectuate an operation on the measurement instrument from a user using a browser executing on a remote client computer. In response, the operation is effectuated in accordance with the commands or information provided. Results of the operation, where applicable, are returned to the browser of the remote client computer for the user. The operation may be any one of a number of monitoring or control operations, and the commands or information facilitate identification of the monitoring or control operation of interest. The results of the operation returned may include data obtained from the measurement instrument.

[0005] In one embodiment, the measurement instrument is web enabled by coupling the measurement instrument to a server computer equipped with a web server, a script executor and one or more web pages having one or more associated scripts to facilitate the interactions between the user using the browser of the remote client computer and the measurement instrument.

[0006] In an alternate embodiment, the measurement instrument is equipped with a web based control subsystem including a web server, a script executor and one or more web pages having one or more associated scripts to facilitate the interactions between the user using the browser of the remote client computer and the measurement instrument.

BRIEF DESCRIPTION OF DRAWINGS

[0007] The present invention will be described by way of exemplary embodiments, but not limitations, illustrated in the accompanying drawings in which like references denote similar elements, and in which:

FIGURE 1 illustrates an overview of the present invention;
FIGURE 2 is a flow diagram illustrating the method steps of the present invention in accordance with one embodiment;
FIGURES 3a-3b illustrate two embodiments of the web enabled measurement instrument of FIGURE 1;
FIGURES 4a-4b illustrate a hardware view and a software view of one embodiment of web based control subsystem of Figure 3b;
FIGURES 5a-5b illustrate two exemplary screens for one embodiment of a web page interface in accordance with the present invention;

DETAILED DESCRIPTION OF THE INVENTION

[0008] In the following description, various aspects of the present invention will be described. Those skilled in the

art will also appreciate that the present invention may be practiced with only some or all aspects of the present invention. For purposes of explanation, specific numbers, materials and configurations are set forth in order to provide a thorough understanding of the present invention. However, it will also be apparent to one skilled in the art that the present invention may be practiced without the specific details. In other instances, well known features are omitted or simplified in order not to obscure the present invention.

[0009] Parts of the description will be presented in terms of operations performed by a computer system, using terms such as data, flags, bits, values, characters, strings, numbers, Boolean operators, etc., consistent with the manner commonly employed by those skilled in the art to convey the substance of their work to others skilled in the art. As well understood by those skilled in the art, these quantities take the form of electrical, magnetic, or optical signals capable of being stored, transferred, combined, and otherwise manipulated through mechanical and electrical components of the computer system; and the term computer system includes general purpose as well as special purpose data processing machines, systems, and the like, that are standalone, adjunct or embedded.

[0010] Various operations will be described as multiple discrete steps in turn in a manner that is most helpful in understanding the present invention, however, the order of description should not be construed as to imply that these operations are necessarily order dependent, in particular, the order of their presentations.

[0011] Referring now to **Figure 1**, a block diagram illustrating an overview of the present invention is shown. As illustrated, in accordance with the present invention, measurement instrument **112** is web enabled, forming web enabled measurement instrument **102** that is capable of receiving commands or information to effectuate an operation on measurement instrument **112** from a user using browser **120** executing on remote client computer **104**. Examples of such operations are monitoring or control operations, and the commands or information are monitoring commands, control commands, or information that facilitate identification of the monitoring or control operation of interest. As will be described in more details below, in response, the operation is effectuated in accordance with the commands or information provided. Furthermore, results of the operation, if applicable, are returned to browser **120** of remote client computer **104** for the user. The results of the operation returned may include data obtained from measurement instrument **112**.

[0012] Client computer **104** including browser **120** are intended to represent a broad category of these elements known in the art. For example, client computer **104** may be model Dimension® desktop computer, available from Dell Computer of Austin Texas, whereas browser **120** may be Navigator® 3.0 available from Netscape Communication of Mountain View, CA. As will be readily apparent from the description to follow, the present invention may be practiced with a basic browser without additional plug-ins, however, does not preclude their use for additional functions.

[0013] Client computer **104** is coupled to web enabled measurement instrument **102** through communication links **108** and **110** and communication network(s) **106**. Communication links **108** and **110** as well as communication network (s) **106** are also intended to represent a broad category of these elements known in the art. For example, communication network(s) **106** may be a local area network (LAN), such as an Ethernet network, formed with hubs and cables, and communication links **108** and **110** are segments of the LAN. Alternatively, communication network(s) **106** may be a private or virtually private wide area network (WAN), such as a frame relay network, formed with router, switches and cables, and communication links **108** and **110** are access lines to the WAN. Communication network(s) **106** may also be the a public network, such as the Internet, formed with point of presence (POP) providers interconnected by backbone interconnections, and optionally the public switching telephone network (PSTN), and communication links **108** and **110** may be networked or dial up links to the POP providers.

[0014] Measurement instrument **102** is web enabled by associating web server **116**, script executor **114** and one or more web pages **118** with measurement instrument **112**. Web pages **118** also include one or more associated scripts (not separately shown). Web server **116** in conjunction with at least one of web pages **118** enable the user to conveniently provide monitoring commands, control commands, or information associated with monitoring or control operations on measurement instrument **112** using commonly available browser **120** from a commonly available remote client computer **104**. Web server **116** in conjunction with script executor **114** and web pages **118** (including associated scripts) enable measurement instrument **112** to be monitored or controlled or data to be extracted responsive to the monitoring commands, control commands or information provided. Furthermore, elements **114-118** enable responses of measurement instrument **112** to the control or the data extracted to be returned to browser **120** and conveniently rendered for the user. Web server **116** and script executor **114** are intended to represent a broad category of these elements known in the art. An example of web server **116** is Quid Pro Quo™ web server, a shareware software available from Social Engineering Inc., of Berkeley, CA, whereas an example of script executor **114** is Frontier Scripting™ avallable from UserLand Software of CA. Other web servers and script executors may be employed. The contents of web pages **118** including their associated scripts are application dependent. In alternate embodiments, in lieu of script executor and scripts, equivalent implementations in other high level languages, such as JAVA™ execution environment and JAVA® applets, may be employed instead.

[0015] Skipping now to **Fig. 5a-5b**, two exemplary screens of one embodiment of a web page interface are illustrated. For the first exemplary screen, interface web page **500a** is a simplified abstraction of the front panel of the measurement instrument, in this case, an oscilloscope. As shown, interface web page **500a** includes display area **502** where the

measurement data are graphically presented. Additionally, interface web page **500a** includes various control buttons **504** to facilitate user selection of different monitoring or control command groups, e.g. commands associated with controlling the vertical setting of the measurement instrument, command associated with controlling the horizontal setting of the measurement instrument, and so forth. For the second exemplary screen, interface web page **500b** provides additional commands or input fields to facilitate further user inputs for parameter values associated with selected command group. As shown, interface web page **500b** still includes display area **502** where the measurement data are graphically presented. However, in lieu of commands **504**, interface web page **500b** includes input fields **506** to facilitate input of various parameter values associated with the monitoring or control operations. Examples of these input fields include the scale, positioning and coupling of a channel, and the source and voltage level of a trigger. In one embodiment, interactions with selected ones of control buttons **504** or input fields **506** result in identifiers of scripts to be executed to be provided to the web server, along with the user supplied parameters values, if any.

[0016] Referring now to **Fig. 2**, the method steps of the present invention in accordance with one embodiment are shown in further details. As illustrated, a user desiring to remotely monitor or control a measurement instrument first connects to a web server associated with the measurement instrument using a browser executing on a remote client computer, step **202**. In response to the connection, the associated web server presents an interface web page to the user, through the connecting browser, step **204**. Next, the user interacts with selected ones of the control buttons or the input fields to provide the measurement instruments with monitoring commands, control commands, or information associated with monitoring or control operations on the measurement instrument, step **206**. At step **208**, the monitoring commands, control commands, or information are sent from the browser to the associated web server. In response to the received monitoring commands, control commands, or information, the web server causes one or more scripts to be launched for execution by the script executor, step **210**. At step **212**, the script executor executes the scripts providing commands to the measurement instrument as appropriate. Examples of such commands, in the case of an oscilloscope, include but not limited to "autoset", "setting the vertical scale to a particular scale value", "setting the horizontal scale to a particular scale value" and "retrieving the x, y values of the displayed graph".

[0017] At step **214**, the measurement instrument responds to the monitoring commands, control commands, or information provided, or the data requested of the script executor. At step **216**, the script executor transfers the responses of the measurement instrument or the data extracted to the web server for forwarding to the user. At step **218**, the web server returns the responses of the measurement instrument or the data extracted to the browser for rendering for the user. Finally, at step **220**, the browser refreshes the interface web page with the responses or data received.

[0018] Steps **208-220** are repeated for as long as the user continues to interact with the interface web page. Eventually, the process terminates when the user disconnects from the web server.

[0019] **Fig. 3a-3b** illustrate two embodiments of web enabled measurement instrument **102**. In **Fig. 3a**, web enabled measurement instrument **102'** is formed by externally coupling measurement instrument **112'** to server computer **130**, which is equipped with web server **116'**, script executor **114'** and one or more web pages **118'**. As described earlier, web pages **118'** include one or more associated scripts. For the illustrated embodiment, measurement instrument **112'** is externally coupled to server computer **130** through communication link **132**. In one embodiment, communication link **132** is a conventional RS232 interface. In alternate embodiments, communication link **132** may be a conventional serial interface or a conventional parallel interface instead. In yet other embodiments, communication link **132** may be a bus interface instead, such as the universal serial bus or the IEEE 1394 high speed serial bus.

[0020] In **Fig. 3b**, web enabled measurement instrument **102"** is formed by closely coupling acquisition subsystem **146** and display subsystem **150** of measurement instrument **112"** to newly provided web based control subsystem **160**. In other words, newly provided web based control subsystem **160** and the conventional subsystems, i.e. subsystems **142-150**, together form an integrated web enabled measurement instrument **102"**. Vertical subsystem **142**, trigger subsystem **144**, acquisition subsystem **146**, horizontal subsystem **148** and display subsystem **150** are intended to represent a broad category of these elements known in the art. Each of elements **142-150** performs its conventional functions, and the elements' constitutions are known. Accordingly, elements **142-150** will not be further described.

[0021] **Fig. 4a-4b** illustrate an hardware and a software view of an exemplary embodiment of web based control subsystem **160**. As shown in **Fig. 4a**, for the illustrated embodiment, web based control subsystem **160'** includes processor **402**, non-volatile storage **404**, memory **406**, bus bridge **408**, and I/O interface **410**, coupled to each other through internal bus **412**. Bus bridge **408** is also coupled to the acquisition subsystem to facilitate provision of monitoring or control commands and receipt of responses of the acquisition subsystem to and from the acquisition subsystem, whereas I/O interface **410** facilitates receipt of the graphical display data from the acquisition subsystem. In one embodiment, internal bus **412** is a peripheral component interconnect (PCI) bus, whereas bus bridge **408** is a PCI-PCI bridge. As shown in **Fig. 4b**, non-volatile storage **404** and memory **406** are used to store the permanent and working copies of the web server, the script executor, and the web pages including the associated scripts described earlier. The permanent copy of the web server etc. may be preloaded in the factory or field installed (if web enabled measurement instrument **102"** also includes the necessary input or communication devices). Elements **402-412** are all intended to represent a broad category of these elements known in the art. Each of elements **402-412** perform its conventional

functions known in the art, and their constitutions are known. Accordingly, elements **402-412** will not be further described either.

[0022] Tables 1-9 illustrate various exemplary scripts suitable for used to practice the present invention. Tables 1-7 form a single script, and are designated a separate tables merely for convenience in referring to the different sections of the script. In particular, Tables 1-7 illustrate an exemplary script suitable for use to generate a "front panel" web page for the user. For the exemplary script, table 1 describes various relevant parameters, e.g. the client's IP address, the server name, the TCP/IP port of the server, and so forth. Table 2 describes the logic associated with the various actions to be taken in response to certain user actions. Table 3 describes the various command "buttons" to be included and displayed for the web page. Tables 4 and 5 illustrates two exemplary scripts for generating enumerated and text choices for input fields of the web page. Table 6 describes the associated logic. Finally, tables 7 and 8 illustrate two exemplary scripts for providing monitoring or control commands to the measurement instrument. Table 7 illustrates the exemplary source code for querying an oscilloscope's current settings, whereas Table 8 illustrates the exemplary source code for capturing the current measurement data. The precise meaning of these statements are unimportant, even though they are readily apparent to those skilled in the art. The exemplary scripts are merely shown for illustrative purpose. The precise statements to be employed are application dependent, i.e. dependent on the type of measurement instrument, the type of end user interface offered, the number and types of user commands and input fields offered, the nature of responses to the various user actions, and so forth. Implementations of these and other application dependent scripts are well within the ability of those skilled in the art. Accordingly, they will not be further described.

Table 1

```
>on frontpanel (adrParams)
<< Available parameters [WebSTAR 1.2]
    << pathArgs - stuff in the URL after a $
    << httpSearchArgs - stuff in the URL after a ?
    << username - authenicated user name
    << password - authenicated password
    << fromUser - not required by http std. e-mail address of remote user
    << clientAddress - IP address or domain name of remote client's host
    << serverName - name or IP address of this server
    << serverPort - TCP/IP port number being used by this server
    << scriptName - URL name of this script
    << contentType - MIME content of post_args
    << referer - the URL of the page referencing this document
    << userAgent - the name and version of the WWW client software being used
    << action - the action name, if the script was an action (or CGI, ACGI, ...)
    << actionPath - the Mac path to the action CGI (this script)
    << postArgs - stuff sent from forms, etc. when POST method is used
    << method - GET, POST, etc.  Used to tell if post_args are valid
    << clientIP - IP address of the client
    << fullRequest - the entire HTTP request received from the client
    << connectionID - unique ID of http connection
    << argTable.xxx - form data when POST method is used, where xxx is the field name
```

Table 2

```
> local (htmltext = webServer.httpHeader ())
> on add (s)
    > htmltext = htmltext +s + cr
> with adrParams^
    > add ("<html>")
    > add ("<head>")
    >add (<title>TekScope Front Panel</title>")
    > add ("</head>")
    > add ("<body>")
    > if defined (argTable.Key)
        > with workspace
            > local
                >comandString=argTable.key
                > errorMessage=""
            > add ("<P>Key: "commandString+"<P>")
            > commandString = ":FPANEL:PRESS "+commandString
            > errorMessage = SendCommandNoOpen(commandString)
            > if (errorMessage ==")
                > add ("(command executed without Frontier errors)<P>")
            > else
                > add ("<P>(command returned error:"+ errorMessage +")<P>")
    > if defined (argTable.Turn)
        > with workspace
            > local
                > commandString =argTableTurn
                > I, numTurns
                > errorMessage = ""
            > add ("<P>Turns: "+commandString+"<P>")
            > commandString = ":FPANEL:TURN "+commandString
            > bundle = Get number of turns
                > local
                    > numLength, numStart
                > numStart = string.patternMatch(",", commandString)+1
                > numLength = string.length(commandString) - numStart+1
                > numTurns = short(string.mid(commandString, numStart, numLength)))
                > numTurns = abs(numTurns)
            > for I = 1 to numTurns
                > errorMessage = SendCommandNoOpen(commandString)
                > if (errorMessage ! ==")
                    > break
            > if (errorMesage ==")
                > add ("command executed without Frontier errors)<P>")
            > else
                > add("<P>(command returned error: " + errorMessage+")<P>")
                > add("<P>(loop Index: "+ I +";NumTurns: ": " +  numTurns +")<P>")
```

Table 3

```
> with workspace
    > local
        > imageName
    > if (string.mid(imageName,1,3)!="ERR")
        > add ("<image src=\"scopeimages/"+imageName+"\">")
    > else
        > add("<P> Error reading image<P>")
> bundle ,, Add form for keypad
    > on addKey (keyString)
        > add("<INPUT TYPE=\"SUBMIT\" NAME=\"Key\"
VALUE=\""+keyString+"\">")
    > on addTurn (turnString)
        > add("<INPUT TYPE=\"SUBMIT\" NAME=\"Turn\"
VALUE=\""+turnString+"\">")
    > add("<FORM ACTION=\"http://128.181.40.171/tekscope.frontpanel.fcgi\"
METHOD=POST>")
    > add("<INPUT TYPE=\"SUBMIT\" NAME=\"Refresh\">")
    > add("<BR>")
    > bundle ,,bezel buttons, clear menu, and toggle
        > addKey("BMenu1")
        > addKey("BMenu2")
        > addKey("BMenu3")
        > addKey("BMenu4")
        > addKey("ClearMenu")
        > addKey("Toggle")
        > add("<BR>")
    > bundle ,,Meter, Hold,Acquire,Meas,Cursors
        > addKey("Meter")
        > addKey("RunStop")
        > addKey("Acquire")
        > addKey("MeasMenu")
        > addKey("Cursors")
        > add("<BR>")
    > bundle ,,Scope, AutoRange, Save/Recall, Display, Utility
        > addKey("Scope")
        > addKey("AutoRange")
        > addKey("SRSetup")
        > addKey("Display")
        > addKey("Utility")
        > add("<BR>")
    > bundle ,,Ch1, Ch2, Math, RefA, RefB, WfmOff
        > addKey("Ch1")
        > addKey("Ch2")
        > addKey("Math")
        > addKey("RefA")
        > addKey("RefB")
        > addKey("WfmOff")
        > add("<BR>")
```

```
> bundle ,,Vert Menu, Horiz Menu, Trigger Menu, Mag, 50%
    > addKey("VertMenu")
    > addKey("HorzMenu")
    > addKey("Mag")
    > addKey("TrigMenu")
    > addKey("SetLevel")
    > add("<BR>")
> bundle ,,Knobs
        > bundle ,,Positive turns
            > addTurn("VertScale,+1")
            > addTurn("Vertpos,+25")
            > addTurn("HorzScale,+1")
            > addTurn("HorzPos,+25")
            > addTurn("TrigLevel,+25")
            > addTurn("GPKnob,+1")
        > add("<BR>")
        > bundle ,,Negitive turns
            > addTurn("VertScale,-1")
            > addTurn("Vertpos,-25")
            > addTurn("HorzScale,-1")
            > addTurn("HorzPos,-25")
            > addTurn("TrigLevel,-25")
            > addTurn("GPKnob,-1")
        > add("<BR>")
    > add("</FORM>")
> add("</body>")
> add("</html>")
> return(htmltext)
```

Table 5

```
> On GenerateSelectTag (theCommand, theName, theOptions)
    > local
        > theTag
        > theCurrentSetting
        > i
    > on add (s)
        > theTag = theTag +s + char(13)
    > theCurrentSetting = workspace.QueryScopeSetting(theCommand)
    > add ("<SELECT NAME=\" =theName + "\" SIZE=1>")
    > local
        > theOptionName
        > theOptionLabel
        > theOptionTag
    > for I =1 to sizeOf(theOptions)
        > theOptionName = theOptions(i)[1]
        > theOptionLabel = theOptions(i)[2]
        > theOptionTag ="<OPTION VALUE=\"" + theOptionName +"\""
        > if (theOptionName= = theCurrentSetting)
              > theOptionTag = theOptionTag + "SELECTED"
        > theOptionTag = theOptionTag + ">" + theOptionLabel


        >add (theOptionTag)
    > add ("</SELECT>")
    >return theTag
>GenerateSelectTag ("CH!:SCA, "Ch1Scale", {{ "5.0E-3", "5 mV"},\
        > {"1.0E-2", "10mV"}, {"2.0E-2", "20 mV"},{"5.0E-2", "50 mV"},\
        > {"1.0E-1", "100mV"}, {"2.0E-1", "200 mV"},{"5.0E-1", "500 mV"},\
        > {"1.0E-0", "1 V"}, {"2.0E-0", "2 V"},{"5.0E-0", "5 V"},\
        > {"1.0E1", "10 V"}, {"2.0E1", "20 V"},{"5.0E1", "50 V"},\
        > })
<<aaaRecordTestScript (ACQ:MODE",{{"SAM", "SAMPLE"},\
    <<{"PEAK", "Peak Detect"}, {"AVE", "Average"}, {"ENV, "Envelope"}})
```

Table 6

```
> On GenerateTextTag (theCommand, theName, theSize)
    > local
        > theTag
        > theCurrentSetting
        > i
    > theCurrentSetting = workspace.QueryScopeSetting(theCommand)
    > theTag="<INPUT TYPE=TEXT SIZE=" +theSize +"Name=\" +theName + "\"VALUE="+\
        > theCurrentSetting + ">"
    > return theTag
> msg(GenerateTextTag (":TRIGGER:MAIN:LEVEL","TrigLevel",8))
```

Table 7

```
> On QueryScopeSetting (settingToQuery)
    > local
        > queryResponse
        > errorMsg
    > if string.nthChar(settingToQuery, string.length(settingToQuery)) !="?"
        > settingToQuery = settingToQuery + "?"
    > with system.extensions.SerialPortCommands, workspace
        > try
            > sendStringSlowly (settingToQuery + ScopeEOL)
            > queryResponse = workspace.ReadSerialLine(ScopeSerialPort,
180)(settingToQuery + ScopeEOL)
            > return queryResponse
        > else

            > errorMsg = "error:+tryError +"Querying: " + settingToQuery +"; Port: " +
ScopeSerialPort
            << msg(errorMsg)
            > scriptError(tryError)]
            >return(errorMsg)
> dialog.Alert("Acquire mode: " + QueryScopeSetting(":ACUIRE:MODE"))
```

Table 8

```
> on GrabNewScreenForFCGI ()
    > with workspace, clip2gif
        > local
```

```
            > gifName, gifPath
        > bundle = Make sure ODB variables are defined
            > if not (defined(workspace.ScopeGIFFolder))
                > workspace.ScopeGIFFolder = "Macintosh HD:MikeGauland:Quid Pro
Quo:scopeimages:"
            > if not (file exists(workspace.ScopeGIFFolder))
                > file.newFolder(workspace.ScopeGIFFolder)
            > if not (defined(workspace.ScopeGIFNumber))
                > workspace.ScopeGIFNumber=0
        > gifName = "screen" +ScopeGIFNumber +"-gif"
        > gifPath = ScopeGIFFolder +gifName
        > ScopeGIFNumber= ScopeGIFNumber+1
        > errorMessage= GrabScreenNoOpen("RAM Disk:scope.tif")
        > if (errorMessage=="")
            > app.start ("clip2gif")
            > app.bringToFront()
            <<save ("RAM Disk:scope.tif", saveIn:" gifPath", as:gif)
            > save (RAM Disk:scope.tif", saveIn:"RAM Disk:scope.gif", as:gif, depth:1)
            > Frontier.bringToFront()
            > local
                > gifData
            > file.open ("RAM Disk:scope.tif")
            > gifData = file.read("RAM Disk:scope.gif", infinity)
            > gifData [19]= 0xff
            > file.close("RAM Disk:scope.gif")
            > gifData [19]= 0xff
            > file.new(gifPath)
            > file.setCreator(gifPath,'c2gf')
            > file.setType(gifPath,'GIFf')
            > file.write(gifPath, gifData)
            > file.setCreator(gifPath,'c2gf')
            > file.setCreator(gifPath,'c2gf')
            > file.close(gifPath)
            > return gifName
        > else
            > return "ERR" +errorMessage
    > GrabNewScreenForFCGI ()
```

[0023] In general, those skilled in the art will appreciate that through the novel approach of web enabling a measurement instrument as described earlier, a user is advantageously enabled to monitor or control the measurement instrument remotely through commonly available tools such as a browser executing on a commonly available client computer, a very user friendly approach. Furthermore, the web enabling approach described earlier may be effectuated through substantially "off-the-shelf" components, without resorting to costly and burdensome proprietary interfaces and mechanisms.

[0024] Those skilled in the art will also recognize that the present invention is not limited by the details nor the embodiments described. Instead, the present invention can be practiced with modifications and alterations within the spirit and scope of the appended claims. The description is thus to be regarded as illustrative instead of restrictive on the present invention.

[0025] Thus, a method and apparatus for remote monitoring or control of a measurement instrument has been de-

scribed.

Claims

1. In a server computer, a machine implemented method comprising the steps of:

 a) providing by a web server to an executor, both executing on the server computer, commands or information, received by the web server from a browser executing on a remote client computer, corresponding to an operation to be effectuated on a measurement instrument; and
 b) effectuating the operation on the measurement instrument by the executor in accordance with the commands or information provided by the web server.

2. The method of claim 1 wherein

 the step of providing comprises receiving by the web server information identifying a collection of executable instructions to be executed to effectuate the operation on the measurement instrument, from the browser of the remote client computer, and providing by the web server to the executor the identified collection of executable instructions; and
 the step of effectuating comprises executing by the executor the identified collection of executable instructions.

3. The method of claim 1 wherein the step of effectuating comprises forwarding a control or data obtaining command to the measurement instrument by the executor of the server computer.

4. The method of claim 1 wherein method further comprises the step of:

 c) forwarding results of the effectuated operation to the browser of the remote client computer by the web server of the server computer.

5. The method of claim 4 wherein the step of forwarding further comprises receiving by the executor responses of the measurement instrument or data being obtained, and providing by the executor to the web server the received responses or data obtained.

6. In a measurement instrument, a method comprising the steps of:

 a) providing control or data obtaining commands by a web based control subsystem to an acquisition subsystem, both subsystems of the measurement instrument, in response to commands or information to effectuate an operation on the measurement instrument receiving by the web based control subsystem from a browser executing on a remote client computer; and
 b) operating the measurement instrument by the acquisition subsystem in accordance with the control or data obtaining commands provided.

7. The method of claim 6 wherein the step of providing comprises receiving by a web server of the web based control subsystem, from the browser of the remote client computer, information identifying a collection of instructions to be executed to effectuate the operation, providing by the web server to an executor of the web based control subsystem the received identification information, and executing the identified collection of instructions by the executor, which results in the provision of the control or data obtaining commands to the acquisition subsystem.

8. The method of claim 6 wherein the method further comprises the step of:

 c) providing results of the operation by the web based control subsystem to the browser of the remote client computer.

9. The method of claim 8 wherein the step of providing comprises receiving by an executor of the web based control subsystem of the measurement instrument responses or data being obtained from the acquisition subsystem of the measurement instrument, providing by the executor to a web server of the web based control subsystem the received responses or data obtained, and forwarding by the web server to the browser of the remote client computer the provided responses or data obtained.

10. A server computer comprising:

a storage medium having stored therein a first and a second plurality of programming instructions that implement a web server and an executor respectively, when executed, enabling the server computer to operate to receive commands or information to effectuate an operation on a measurement instrument from a browser of a remote client computer, enabling the server computer to operate the measurement instrument responsive to the received commands or information, and forwarding results of the operation, when applicable, to the browser of the remote client computer; and
(b) a processor coupled to the storage medium to execute the first and second plurality of programming instructions.

11. The server computer of claim 10 wherein, when executed, the first plurality of programming instructions implementing the web server enable the server computer to receive information corresponding to a third plurality of programming instructions to be executed to effectuate the operation on the measurement instrument from the browser of the remote client computer, and provide the received information to the executor.

12. The server computer of claim 10 wherein, when executed, the second plurality of programming instructions implementing the executor enable the server computer to execute a third plurality of programming instructions to effectuate the operation on the measurement instrument, including forwarding of a control command to the measurement instrument.

13. The server computer of claim 10 wherein, when executed, the second plurality of programming instructions implementing the executor enable the server computer to execute a third plurality of programming instructions to effectuate the operation on the measurement instrument, including forwarding of a data obtaining command to the measurement instrument, receiving the data being obtained from the measurement instrument, and providing the received data to the web server for forwarding to the browser of the remote client computer.

14. The server computer of claim 10 wherein the server computer further comprises a first and a second communication interface coupling the remote client computer and the measurement instrument to the server computer respectively.

15. A measurement instrument comprising:

an acquisition subsystem to be used to acquire measurement data; and
a web based control subsystem coupled to the acquisition subsystem to be used to receive commands or information corresponding to an operation from a browser executing on a remote client computer, and to cause the acquisition subsystem to effectuate the operation responsive to the received commands or information.

16. The measurement instruction of claim 15 wherein the web based control subsystem comprises:

a storage medium having stored therein a first plurality of programming instructions implementing a web server, when executed, enabling the web based control subsystem to receive from the browser of the remote client computer information corresponding to a second plurality of programming instructions to be executed to cause the operation to be effectuated by the acquisition subsystem, and when applicable, to forward responses of the acquisition subsystem to the browser of the remote client computer; and
a processor coupled to the storage medium to be used to execute the first plurality of programming instructions.

17. The measurement instrument of claim 16 wherein the responses of the acquisition subsystem forwarded include data obtained from the acquisition subsystem.

18. The measurement instruction of claim 15 wherein the web based control subsystem comprises:

a storage medium having stored therein a first plurality of programming instructions implementing an executor, when executed, enabling the web based control subsystem to execute a second plurality of programming instructions to cause the acquisition subsystem to effectuate the operation on the measurement instrument, and when applicable, receiving responses of the acquisition subsystem to be forwarded to the browser of the remote client computer; and
a processor coupled to the storage medium to be used to execute the first plurality of programming instructions.

19. The measurement instrument of claim 18 wherein the storage medium further having stored therein the second plurality of programming instructions, and the processor is also used to execute the second plurality of programming instructions.

20. The measurement instrument of claim 18 wherein when executed, the second plurality of programming instructions cause a control of data obtaining command to be provided from the web based control subsystem to the acquisition subsystem.

21. A computer program for implementing a method as claimed in any one of claims 1 to 9.

22. A computer-readable storage medium storing a program as claimed in claim 21.

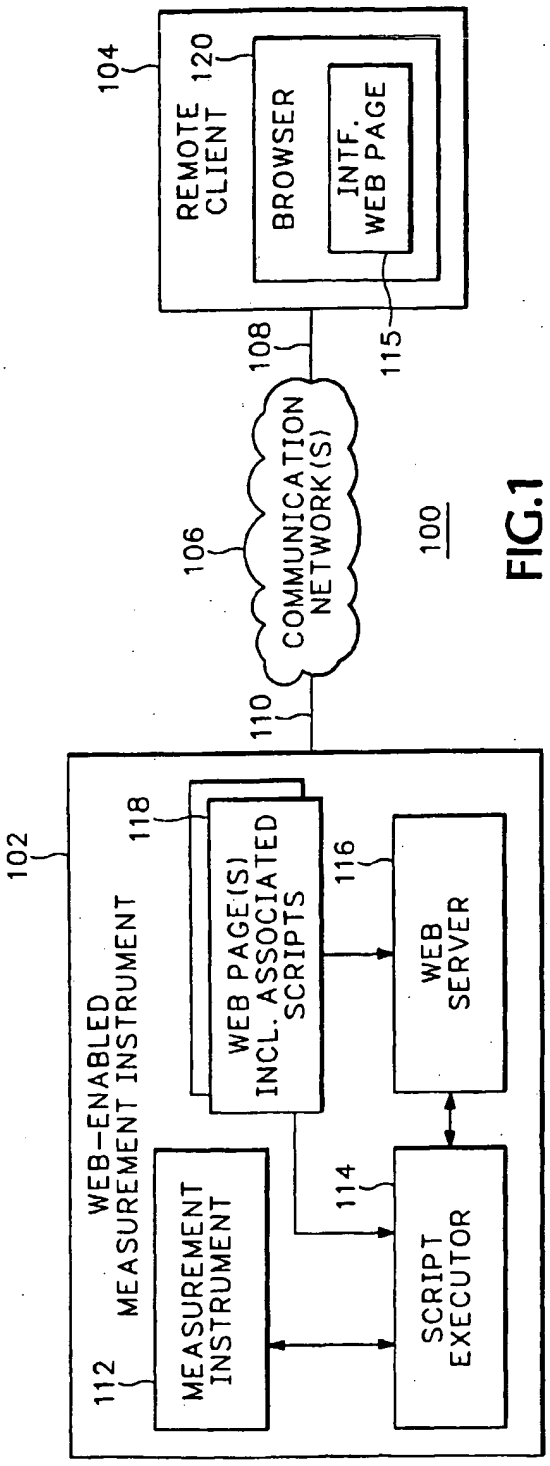

104
REMOTE CLIENT
120
BROWSER
INTF. WEB PAGE
115
108
106
COMMUNICATION NETWORK(S)
110
100
102
WEB-ENABLED MEASUREMENT INSTRUMENT
118
WEB PAGE(S) INCL. ASSOCIATED SCRIPTS
116
WEB SERVER
112
MEASUREMENT INSTRUMENT
114
SCRIPT EXECUTOR

FIG.1

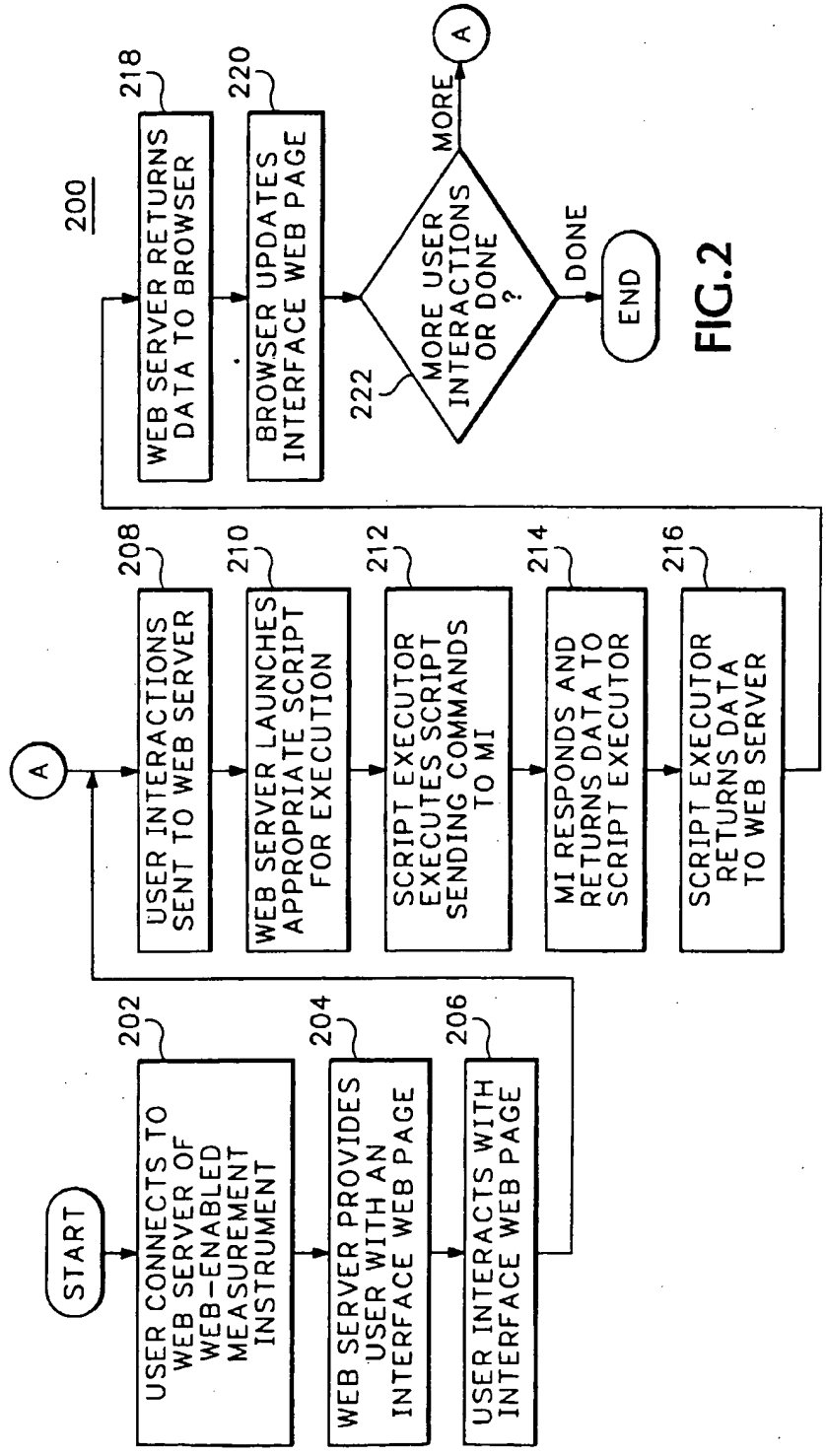

200
START
USER CONNECTS TO WEB SERVER OF WEB-ENABLED MEASUREMENT INSTRUMENT
202
WEB SERVER PROVIDES USER WITH AN INTERFACE WEB PAGE
204
USER INTERACTS WITH INTERFACE WEB PAGE
206
A
USER INTERACTIONS SENT TO WEB SERVER
208
WEB SERVER LAUNCHES APPROPRIATE SCRIPT FOR EXECUTION
210
SCRIPT EXECUTOR EXECUTES SCRIPT SENDING COMMANDS TO MI
212
MI RESPONDS AND RETURNS DATA TO SCRIPT EXECUTOR
214
SCRIPT EXECUTOR RETURNS DATA TO WEB SERVER
216
WEB SERVER RETURNS DATA TO BROWSER
218
BROWSER UPDATES INTERFACE WEB PAGE
220
MORE USER INTERACTIONS OR DONE ?
222
MORE
A
DONE
END

FIG.2

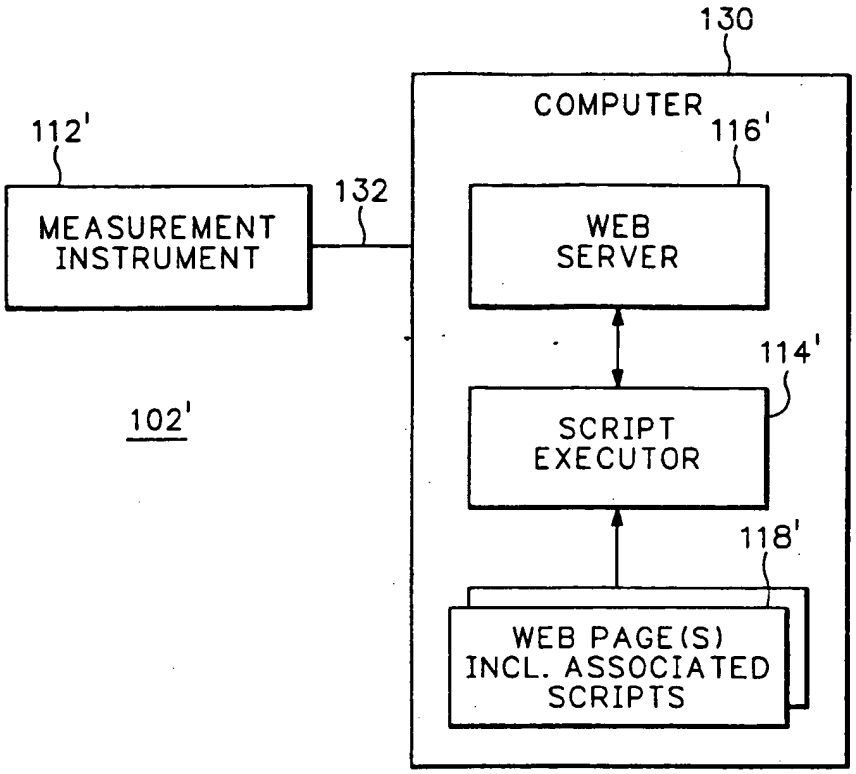

130
COMPUTER
112'
116'
MEASUREMENT INSTRUMENT
132
WEB SERVER
114'
SCRIPT EXECUTOR
102'
118'
WEB PAGE(S) INCL. ASSOCIATED SCRIPTS

FIG.3a

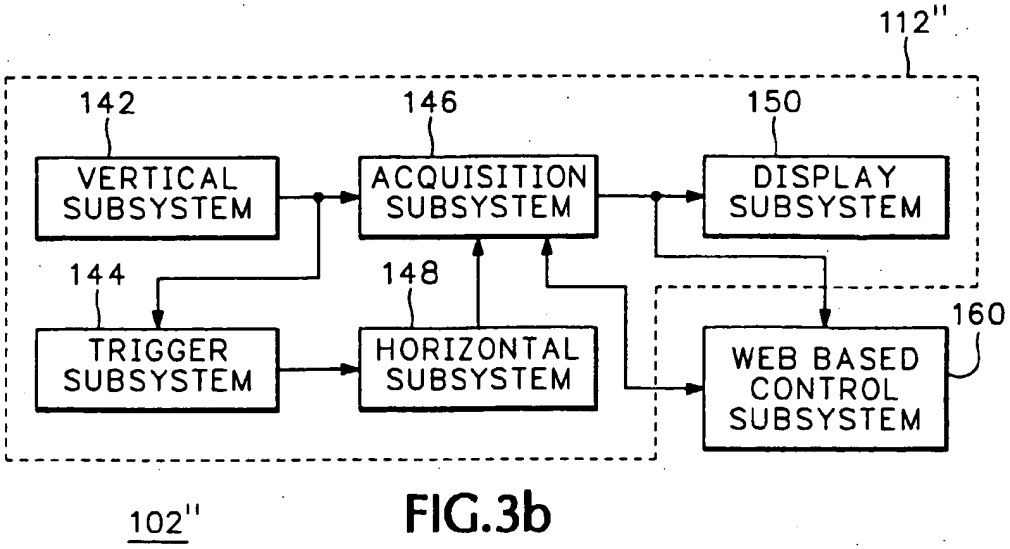

112''
142
146
150
VERTICAL SUBSYSTEM
ACQUISITION SUBSYSTEM
DISPLAY SUBSYSTEM
144
148
160
TRIGGER SUBSYSTEM
HORIZONTAL SUBSYSTEM
WEB BASED CONTROL SUBSYSTEM
102''

FIG.3b

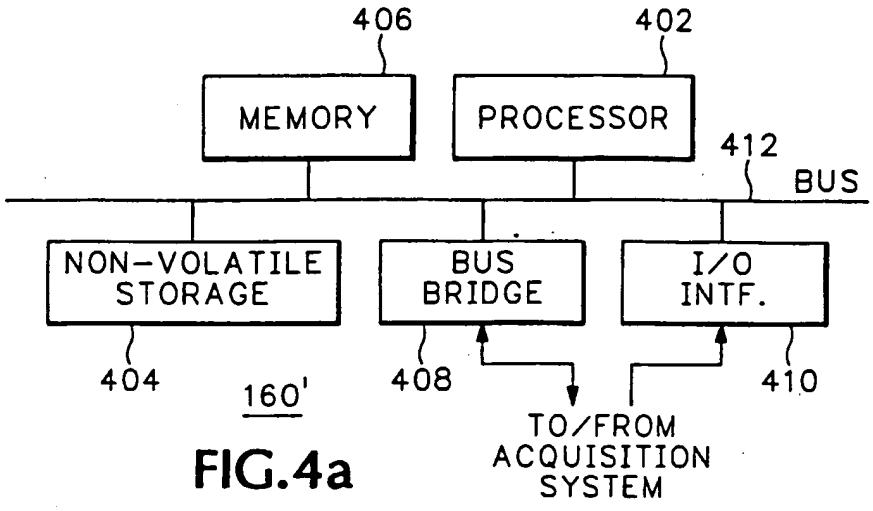

406
402
MEMORY
PROCESSOR
412
BUS
NON-VOLATILE STORAGE
BUS BRIDGE
I/O INTF.
404
160'
408
410
TO/FROM ACQUISITION SYSTEM

FIG.4a

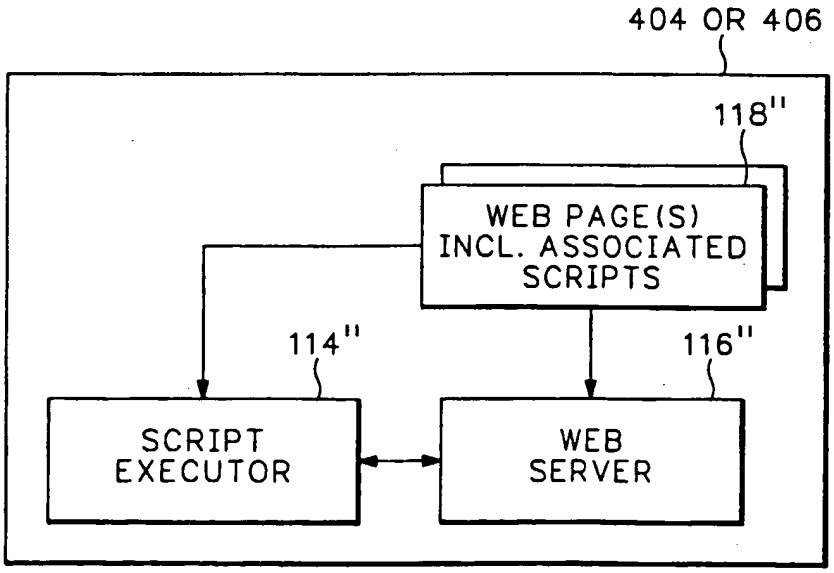

404 OR 406
118''
WEB PAGE(S) INCL. ASSOCIATED SCRIPTS
114''
116''
SCRIPT EXECUTOR
WEB SERVER

FIG.4b

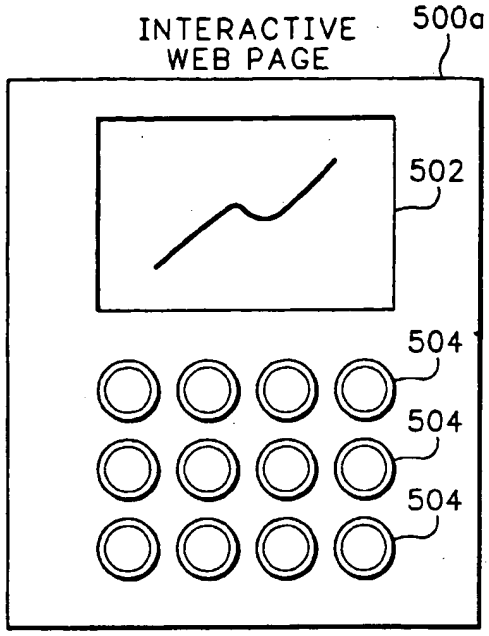

INTERACTIVE
WEB PAGE
500a
502
504
504
504

FIG.5a

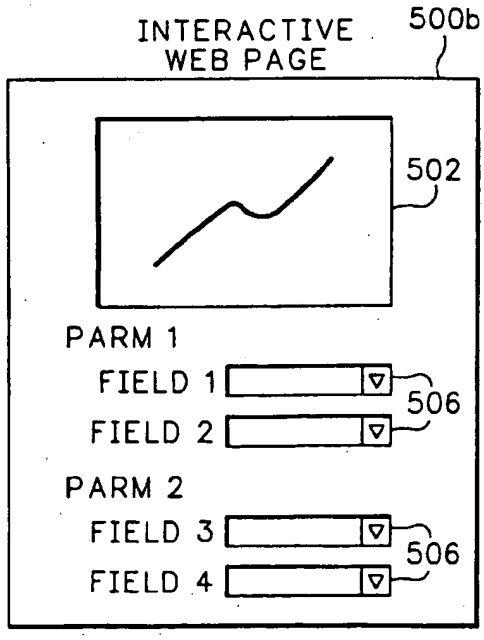

INTERACTIVE
WEB PAGE
500b
502
PARM 1
FIELD 1
FIELD 2
506
PARM 2
FIELD 3
FIELD 4
506

FIG.5b